REMARKS

Claims 1 – 20 remain pending in the present Application.

Response to Arguments Section

The Response to Arguments Section acknowledges the Applicant's arguments with regards to the Schieve and Skrovan references were persuasive. Applicant respectfully reasserts the argument that the present invention is not taught by the Shieve and Skrovan references.

102 Rejections

Claims 1 – 3 and 7 are rejected under 35 U.S.C. 102(b) as being anticipated by Schieve (US Patent 6,018,808). Applicant respectfully asserts that the present invention as recited in Claims 1 – 3 and 7 are neither shown nor suggested by the Schieve reference.

Applicant respectfully asserts that the Schieve reference is not directed to the present invention as recited in Claim 1. Specifically the present invention, as set forth in independent Claim 1 recites in part:

> ... processor configured to ... perform multi-tasking operations while accessing serial presence detect information during boot up operations prior to completing volatile memory initialization.

To the extent the Schieve reference may mention a master handler causes the storing of pertinent information relative to that on which {sic} a microprocessor had been working when the interrupt occurred [Col. 4, lines 40 - 44], Applicant respectfully asserts the Schieve reference does not teach performing ***multi-tasking*** operations ***while accessing serial presence detect*** information during boot up operations. In the Response to Arguments Section, the present Office Action seems to be alleging that the pertinent information relative to that on which microprocessor 10 had been working {sic} when the interrupt occurred [Col. 4, lines 15 – 55] is the same as performing multi-tasking operation while accessing serial presence detect data in an interrupt mode. To the extent the Schieve reference may mention information relative to that on which a microprocessor had been working [Col. 4, lines 15 – 55], Applicant respectfully asserts the Shieve reference does not teach accessing serial presence detect data. In addition, the extent the Shieve reference may mention storing of pertinent information in a

register in the ***absence of RAM*** [Col. 4 lines 31 - 55], Applicant respectfully asserts the Schieve reference arguably ***teaches away*** from ***serial presence detect data.***

To the extent the Schieve reference may mention a master handler causes the storing of pertinent information relative to that on which {sic} a microprocessor had been working when the interrupt occurred [Col. 4, lines 40-44], Applicant respectfully asserts the Schieve reference does not teach performing ***multi-tasking*** operations while ***accessing serial presence detect*** information during boot up operations. For example, the present Application indicates in one exemplary implementation the interrupt driven mode permits the system ***to begin a fetch*** operation and ***proceed with other tasks*** while waiting for the information to be returned [page 8 paragraph 17]. The present Application also ***includes exemplary multi-tasking operations*** that can be performed before completing the initialization of a volatile memory, including for example, a system can initialize a chipset, put static information in registers, initialize GPIOs, initialize a DMA controller, initialize a timer controller, etc. [page 7 paragraph 15].

Applicant respectfully asserts Claims 1 - 7 are allowable as depending from an allowable independent Claim.

With respect to Claim 5, to the extent the Schieve reference may mention a master handler causes the storing of pertinent information relative to that on which {sic} a microprocessor had been working when the interrupt occurred and this storage is ***done in general purpose register 11*** [Figure Col. 4, lines 40 - 45], Applicant respectfully asserts the Schieve reference does not teach wherein said ***system management bus communicates*** serial presence detect data in accordance with directions from a ***system management bus controller*** operating in an interrupt driven mode. In addition, to the extent the Schieve reference may mention this storage is ***done in general purpose register 11*** [Figure Col. 4, lines 40 - 45] and Figure 1 shows the general purpose register 11 is within processor 10, Applicant respectfully asserts the Schieve reference ***teaches away*** from a ***system management bus communicates*** because the information processor 10 was working on would ***remain within*** the processor 10 to store it in register 11 and ***not be communicated out*** on bus 16.

With respect to Claim 6, to the extent the Schieve reference may mention a ROM diagnostic interrupt ***vector table*** having ***addresses entered*** [Col. 4 lines 5 – 10], Applicant respectfully asserts the Schieve reference does not teach serial presence detect data includes ***memory description information***. To the extent the Schieve reference may mention that a vector table address specifies a location where an ***interrupt handler*** may

be found [Col. 2 lines 15 – 17], Applicant respectfully asserts the Schieve reference does not teach serial presence detect data includes ***memory*** description information.

103 REJECTIONS

The present Office Action indicates Claims 8 -20 are rejected under 35 U.S.C. 103 (a) as being unpatentable over Schieve (US Patent 6,018,808) in view of Skrovan et al. (US Patent 6,016,554). Applicant respectfully asserts that the present invention is neither shown nor suggested by the Schieve nor Skrovan et al. references, alone or together in combination.

With respect to Claim 8, to the extent the Schieve reference may mention a master handler causes the storing of pertinent information relative to that on which {sic} a microprocessor had been working when the interrupt occurred [Col. 4, lines 40 - 44], Applicant respectfully asserts the Schieve reference does not teach operating said system management bus controller in a ***multitasking*** environment in which said system management bus controller operates in an interrupt driven mode prior to completing volatile memory initialization, wherein said operating said system management bus controller includes ***retrieving serial presence detect data***. In addition, the present Office Action acknowledges the Schieve reference does not teach programming the system

management bus controller. Applicant respectfully asserts the Skrovan et al. reference does not overcome these and other shortcomings of the Schieve reference.

To the extent the Skrovan et al. reference may mention and show a model of a control signal generator 24 of a model of a ***memory*** controller [Figure 2 [Col. 7 line 65], Applicant respectfully asserts the Skrovan et al. reference does not teach programming the system management ***bus*** controller. In addition, to the extent that the Skrovan et al. reference may mention the model of a control signal is generated to ***test the ability*** a model of a ***microprocessor to respond*** to the model of the control signal [Col. 3 lines 7 – 10, Col. 3 lines 32 – 35], Applicant respectfully asserts the Skrovan et al. reference does not teach programming the system management ***bus*** controller.

In addition, to the extent the Skrovan reference may mention a microprocessor memory and bus ***models*** implemented in ***software*** [Col. 3, lines 9 – 11 and 42 - 50], Applicant respectfully asserts the Skrovan et al. reference does not teach ***operating*** a system management bus controller in an interrupt driven mode in a multi-tasking environment. Furthermore, to the extent the Skorvan et al. reference may mention the ***software elements*** including memory model 16 [Col. 4 lines 3 – 5, Figure 3] which is includes memory control unit 20 [Figure 2] which in turn includes control signal generator 24 [Figure 2] are contained within a ***memory unit*** 44 of a microprocessor

testing system 34 configured to perform functional testing upon the software implementation [Col. 8 lines 65 – 67], Applicant respectfully assert the Skorvan does not teach boot up operations. Applicant respectfully asserts software models running on a system 34 with a microprocessor 36 and memory 44 that have ***already booted up*** do not teach boot up operations.

Furthermore, to the extent the Skorvan reference may show the models within a memory 44 that is ***already available*** for utilization in software implementation [Col. 8 lines 65 – 77], Applicant respectfully asserts the Skorvan reference teaches away from operating in an interrupt driven mode ***prior to completing*** volatile ***memory initialization.*** To the extent the Skorvan may show the memory 44 must be initialized and available to support implementation of the control signal generator model 24, Applicant respectfully asserts the Skorvan reference ***teaches away*** from for the control signal generator acting as a system management bus controller that is programmable and operating the system management bus controller prior to completing volatile memory initialization.

Applicant respectfully asserts Claims 8 - 17 are allowable as depending from an allowable independent Claim 8.

With respect to Claim 14, to the extent the Skorvan reference may mention a control signal generator model within a memory model is configurable to generate a chosen signal a selectrabl number of clock cycles following detection of a trigger event [Col.7 lines 64 – Col. 8 line 3], Applicant respectfully asserts the Skorvan reference does not teach programming of said system management bus controller includes initializing said system management bus controller.

With respect to Claim 15, to the extent the Skorvan reference may mention a flag bit of the control signal generator model is set to "0"preferably by reading a value stored within a status register [Col9 lines 56 -59], Applicant respectfully asserts the Skorvan reference does not teach system management bus programming includes slamming system management bus resource addresses.

With respect to Claim 16, to the extent the Skorvan reference may mention the control signal generator is configuratble to generate a chose clock signal a selectable number or system clock cycles following detection of a trigger event [Col. 4 lines 40 – 44], Applicant respectfully asserts the Skorvan reference does not teach multi-tasking operations are executed while processes for retrieving said serial presence detect data are performed.

With respect to Claim 17, to the extent the Skorvan reference may mention the control signal generator is configuratble to generate a chose clock signal a selectable number or system clock cycles following detection of a trigger event [Col. 4 lines 31 – 48], Applicant respectfully asserts the Skorvan reference does not teach providing a location where serial presence detect data is located; retrieving said serial presence detect data in an interrupt driven mode; performing multi-tasking operations while waiting for said serial presence detect data to be retrieved; and generating an interrupt when said serial presence detect data is retrieved.

With respect to Claim 18, to the extent the Schieve reference may mention a device 42 [Fig. 2], Applicant respectfully asserts the Schieve reference does not teach a display. To the extent the Schieve refrence may show device 42 coupled to IRQ line 40 and ROM 20 coupled to bus controller 16 which is coupled to bus 16 [Fig. 1 and 2], Applicant respectfully assert the Schieve reference does not teach a display device and a non-volatile memory unit coupled to said bus. To the extent the Schieve reference may mention a master handler causes the storing of pertinent information relative to that on which {sic} a microprocessor had been working when the interrupt occurred [Col. 4, lines 40 - 44], Applicant respectfully asserts the Schieve reference does not teach a processor for executing a method of multitasking boot up initialization processes and

Applicant also respectfully asserts the Schieve reference does not teach retrieving serial presence detect data.

Applicant respectfully asserts that the Schieve reference is not directed to the present invention as recited in Claim 18. As set forth above, Applicant respectfully asserts that the present invention is neither shown nor suggested by the Schieve reference. The present Office Action acknowledges the Schieve reference fails to disclose programming a system management bus controller, and also fails to teach operating the system management bus controller in a multi-tasking environment in which the system management bus controller operates in an interrupt driven mode prior to completing volatile memory initialization, wherein operating the system management bus controller includes retrieving serial presence detect data.

Applicant respectfully asserts the Skrovan reference does not overcome these and other shortcomings of the Schieve reference. As set forth above, to the extent the Skrovan et al. reference may mention a memory model implemented in software includes a memory control unit which includes a control signal generator 24 [Col. 3, lines 9-11], Applicant respectfully asserts the Skrovan et al. reference does not teach operating a system management bus controller in an interrupt driven mode in a multi-

tasking environment. Applicant respectfully asserts software models running on a system that has already booted up do not teach boot up operations.

Applicant respectfully asserts Claims 19 - 20 are allowable as depending from an allowable independent Claim 18.

With respect to Claim 19, to the extent the Schieve reference may mention a master handler causes the storing of pertinent information relative to that on which {sic} a microprocessor had been working when the interrupt occurred [Col. 4, lines 40 - 44], Applicant respectfully asserts the Scheive reference does not teach providing a location where serial presence detect data is located; retrieving said serial presence detect data in an interrupt driven mode; performing multi-tasking operations while waiting for said serial presence detect data to be retrieved; and generating an interrupt when said serial presence detect data is retrieved.

With respect to Claim 20, to the extent the Schieve reference may mention a master handler causes the storing of pertinent information relative to that on which {sic} a microprocessor had been working when the interrupt occurred [Col. 4, lines 40 - 44], Applicant respectfully asserts the Schieve reference does not teach temporarily storing serial presence detect data in a processor cache.

CONCLUSION

In light of the above-listed amendments and remarks, Applicant respectfully requests allowance of the remaining Claims. The examiner is urged to contact Applicant's undersigned representative if the Examiner believes such action would expedite resolution of the present Application.

Respectfully submitted,

MURABITO, HAO & BARNES LLP

Date: 10/3, 2007

John F. Ryan
Reg. No. 47,050

Two North Market Street
Third Floor
San Jose, CA 95113
(408) 938-9060